```
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEE              RRR   RRR        FFF
EEE              RRR   RRR        FFF
EEE              RRR   RRR        FFF
EEE              RRR      RRR     FFF
EEE              RRR      RRR     FFF
EEE              RRR      RRR     FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
```

FILEID**DUP3271

```
DDDDDDDD    UU      UU  PPPPPPPP      333333      222222    77777777        11
DDDDDDDD    UU      UU  PPPPPPPP      333333      222222    77777777        11
DD      DD  UU      UU  PP      PP  33      33  22      22          77    1111
DD      DD  UU      UU  PP      PP  33      33  22      22          77    1111
DD      DD  UU      UU  PP      PP          33          22          77      11
DD      DD  UU      UU  PP      PP          33          22          77      11
DD      DD  UU      UU  PPPPPPPP          33          22          77        11
DD      DD  UU      UU  PPPPPPPP          33          22          77        11
DD      DD  UU      UU  PP                  33      22          77          11
DD      DD  UU      UU  PP                  33      22          77          11
DD      DD  UU      UU  PP          33      33    22          77            11          ....
DD      DD  UU      UU  PP          33      33    22          77            11          ....
DDDDDDDD    UUUUUUUUUU  PP            333333    2222222222  77            111111        ....
DDDDDDDD    UUUUUUUUUU  PP            333333    2222222222  77            111111        ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
      SUBROUTINE DUP3271 (lun)
C
C Version:      'V04-000'
C
C*****************************************************************************
C*                                                                           *
C*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
C*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
C*  ALL RIGHTS RESERVED.                                                     *
C*                                                                           *
C*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
C*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
C*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
C*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
C*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
C*  TRANSFERRED.                                                             *
C*                                                                           *
C*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
C*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
C*  CORPORATION.                                                             *
C*                                                                           *
C*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
C*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
C*                                                                           *
C*                                                                           *
C*****************************************************************************
C
C
C
C       Author: Sharon Reynolds                 Creation date:10-Sept-80
C
C
C++
C       Functional description:
C
C       This module builds the error log report for the Dup11/3271
C       Protocol Emulator (XKDRIVER).
C
C       Modified by:
C
C       V03-003 SAR0221         Sharon A. Reynolds,     28-Mar-1984
C               Changed the call to UCB$L_OWNUIC to ORB$L_OWNER.
C
C       V03-002 SAR0113         Sharon A. Reynolds,     23-Jun-1983
C               Changed the carriage control in the 'format' statements
C               for use with ERF.
C
C       V03-001 SAR0034         Sharon A. Reynolds,     8-Jun-1983
C               Removed brief support.
C
C       v02-003 BP0003          Brian Porter,           23-NOV-1981
C               Minor edit.
C
C       v02-002 BP0002          Brian Porter,           06-NOV-1981
C               Added 'device attention' support.
C
C       v02-001 BP0001          Brian Porter,           29-JUN-1981
```

```
C                Added call to logger.
C**
C--

        Include 'SRC$:MSGHDR.FOR /NOLIST'
        Include 'SRC$:DEVERR.FOR /NOLIST'



        Byte            lun

        Parameter       timeout = 96
        Parameter       count = 18

        Integer*4       field
        Integer*4       compress4
        Integer*4       compressC
        Integer*4       rxcsr
        Integer*4       rxdbuf
        Integer*4       parcsr
        Integer*4       txcsr
        Integer*4       txdbuf
        Integer*4       stats(0:18)
        Integer*4       tnak
        Integer*4       rnak
        Integer*4       tenq
        Integer*4       reng
        Integer*4       trvi
        Integer*4       rrvi
        Integer*4       rcnv
        Integer*4       chai
        Integer*4       gpol
        Integer*4       xack
        Integer*4       habo
        Integer*4       dabo
        Integer*4       rtim
        Integer*4       disc
        Integer*4       hard
        Integer*4       tblk
        Integer*4       rblk
        Integer*4       habx
        Integer*4       teot

C
C       Make the contents of the register save area in the error log
C       buffer available to this module
C

        Equivalence     (stats(0),tnak)
        Equivalence     (parcsr,EMB$L_DV_REGSAV(1))
        Equivalence     (rxcsr,EMB$L_DV_REGSAV(2))
        Equivalence     (rxdbuf,EMB$L_DV_REGSAV(3))
        Equivalence     (txcsr,EMB$L_DV_REGSAV(4))
        Equivalence     (txdbuf,EMB$L_DV_REGSAV(5))
        Equivalence     (tnak,EMB$L_DV_REGSAV(6))
        Equivalence     (rnak,EMB$L_DV_REGSAV(7))
```

```
            Equivalence     (tenq,EMB$L_DV_REGSAV(8))
            Equivalence     (renq,EMB$L_DV_REGSAV(9))
            Equivalence     (trvi,EMB$L_DV_REGSAV(10))
            Equivalence     (rrvi,EMB$L_DV_REGSAV(11))
            Equivalence     (rcnv,EMB$L_DV_REGSAV(12))
            Equivalence     (chai,EMB$L_DV_REGSAV(13))
            Equivalence     (gpol,EMB$L_DV_REGSAV(14))
            Equivalence     (xack,EMB$L_DV_REGSAV(15))
            Equivalence     (habo,EMB$L_DV_REGSAV(16))
            Equivalence     (dabo,EMB$L_DV_REGSAV(17))
            Equivalence     (rtim,EMB$L_DV_REGSAV(18))
            Equivalence     (disc,EMB$L_DV_REGSAV(19))
            Equivalence     (hard,EMB$L_DV_REGSAV(20))
            Equivalence     (tblk,EMB$L_DV_REGSAV(21))
            Equivalence     (rblk,EMB$L_DV_REGSAV(22))
            Equivalence     (habx,EMB$L_DV_REGSAV(23))
            Equivalence     (teot,EMB$L_DV_REGSAV(24))


C
C           Define driver offsets for PROTOCOL STATISTICS
C
            Character*11    offset(0:18)
            Data    offset(0)       /'LCB$W_TNAK*'/
            Data    offset(1)       /'LCB$W_RNAK*'/
            Data    offset(2)       /'LCB$W_TENQ*'/
            Data    offset(3)       /'LCB$W_RENQ*'/
            Data    offset(4)       /'LCB$W_TRVI*'/
            Data    offset(5)       /'LCB$W_RRVI*'/
            Data    offset(6)       /'LCB$W_RCNV*'/
            Data    offset(7)       /'LCB$W_CHAI*'/
            Data    offset(8)       /'LCB$W_GPOL*'/
            Data    offset(9)       /'LCB$W_XACK*'/
            Data    offset(10)      /'LCB$W_HABO*'/
            Data    offset(11)      /'LCB$W_DABO*'/
            Data    offset(12)      /'LCB$W_RTIM*'/
            Data    offset(13)      /'LCB$W_DISC*'/
            Data    offset(14)      /'LCB$W_HARD*'/
            Data    offset(15)      /'LCB$W_TBLK*'/
            Data    offset(16)      /'LCB$W_RBLK*'/
            Data    offset(17)      /'LCB$W_HABX*'/
            Data    offset(18)      /'LCB$W_TEOT*'/


C
C           Define descriptions for PROTOCOL STATISTICS
C
            Character*32    descrp(0:18)
            Data    descrp(0)       /'NAK''S TRANSMITTED = *'/
            Data    descrp(1)       /'NAK''S RECEIVED = *'/
            Data    descrp(2)       /'ENQ''S TRANSMITTED = *'/
            Data    descrp(3)       /'ENQ''S RECEIVED = *'/
            Data    descrp(4)       /'RVI''S TRANSMITTED = *'/
            Data    descrp(5)       /'RVI''S RECEIVED = *'/
            Data    descrp(6)       /'CONVERSATIONAL REPLIES IN = *'/
            Data    descrp(7)       /'CHAINED WRITES RECEIVED = *'/
            Data    descrp(8)       /'GENERAL POLLS RECEIVED = *'/
```

```
            Data    descrp(9)       /'INVALID ACK''S RECEIVED = *'/
            Data    descrp(10)      /'HOST ABORTED XMISSIONS = *'/
            Data    descrp(11)      /'DRIVER ABORTED XMISSIONS = *'/
            Data    descrp(12)      /'XMISSION RESPONSE TIMEOUTS = *'/
            Data    descrp(13)      /'DETECTED LINE DISCONNECTS = *'/
            Data    descrp(14)      /'DETECTED HARDWARE ERRORS = *'/
            Data    descrp(15)      /'USER BLOCKS TRANSMITTED = *'/
            Data    descrp(16)      /'USER BLOCKS RECEIVED = *'/
            Data    descrp(17)      /'AMBIGUOUS ACK''S SENT = *'/
            Data    descrp(18)      /'IMPLIED ACK''S SENT (EOT) = *'/


C
C           Contruct the report header
C

            Call FRCTOF (lun)

            call dhead1 (lun,'UBA DUP11')

C
C           Decode and output the DUP11 registers
C

            Call DUP11 (lun,parcsr,rxcsr,rxdbuf,txcsr,txdbuf)

C
C           Output PROTOCOL STATISTICAL INFORMATION for report
C

            Call LINCHK (lun,3)

            Write (lun,5)
5           Format (/' ','PROTOCOL STATISTICAL INFORMATION',/)


            Do 20 I=0,count

            Call LINCHK (lun,2)

            Field=LIB$EXTZV (0,16,stats(I))

            Write (lun,10) offset(I),stats(I),descrp(I),field
10          Format (' ',T8,A<compressC (offset(I))>,T24,Z8.4,/,
            1 T40,A<compressC (descrp(I))>,I<compress4 (field)>,'.')


20          Continue


C
C           Decode and output SOFTWARE INFORMATION for report
C

            call linchk (lun,1)

            write(lun,25)
```

```
25      format(' ',:)

        if (emb$w_hd_entry .ne. 98) then

        call ucb$b_ertcnt (lun,emb$b_dv_ertcnt)

        call ucb$b_ertmax (lun,emb$b_dv_ertmax)
        endif

        call orb$l_owner (lun,emb$l_dv_ownuic)

        call ucb$l_char (lun,emb$l_dv_char)

        call ucb$w_sts (lun,emb$w_dv_sts)

        call ucb$l_opcnt (lun,emb$l_dv_opcnt)

        call ucb$w_errcnt (lun,emb$w_dv_errcnt)

        if (emb$w_hd_entry .ne. 98) then

        call linchk (lun,1)

        write(lun,25)

        call dup3271_qio (lun,emb$w_dv_func)

        call irp$w_bcnt (lun,emb$w_dv_bcnt)

        call irp$w_boff (lun,emb$w_dv_boff)

        call irp$l_pid (lun,emb$l_dv_rqpid)

        call irp$q_iosb (lun,emb$l_dv_iosb1)
        endif

        Return
        END
```

PROGRAM SECTIONS

	Name	Bytes	Attributes
0	$CODE	598	PIC CON REL LCL SHR EXE RD NOWRT LONG
1	$PDATA	110	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2	$LOCAL	1136	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3	EMB	512	PIC OVR REL GBL SHR NOEXE RD WRT LONG
	Total Space Allocated	2356	

ENTRY POINTS

Address	Type	Name
0-00000000		DUP3271

VARIABLES

Address	Type	Name	Address	Type	Name
3-00000086	I*4	CHAI	3-00000096	I*4	DABO
3-0000009E	I*4	DISC	3-0000001C	L*1	EMB$B_DV_CLASS
3-00000010	L*1	EMB$B_DV_ERTCNT	3-00000011	L*1	EMB$B_DV_ERTMAX
3-0000003E	L*1	EMB$B_DV_NAMLNG	3-0000003A	L*1	EMB$B_DV_SLAVE
3-0000001D	L*1	EMB$B_DV_TYPE	3-00000036	I*4	EMB$L_DV_CHAR
3-00000012	I*4	EMB$L_DV_IOSB1	3-00000016	I*4	EMB$L_DV_IOSB2
3-00000026	I*4	EMB$L_DV_MEDIA	3-0000004E	I*4	EMB$L_DV_NUMREG
3-0000002E	I*4	EMB$L_DV_OPCNT	3-00000032	I*4	EMB$L_DV_OWNUIC
3-0000001E	I*4	EMB$L_DV_RQPID	3-00000000	I*4	EMB$L_HD_SID
3-0000003F	CHAR	EMB$T_DV_NAME	3-00000024	I*2	EMB$W_DV_BCNT
3-00000022	I*2	EMB$W_DV_BOFF	3-0000002C	I*2	EMB$W_DV_ERRCNT
3-0000003C	I*2	EMB$W_DV_FUNC	3-0000001A	I*2	EMB$W_DV_STS
3-0000002A	I*2	EMB$W_DV_UNIT	3-00000004	I*2	EMB$W_HD_ENTRY
3-0000000E	I*2	EMB$W_HD_ERRSEQ	2-00000334	I*4	FIELD
3-0000008A	I*4	GPOL	3-00000092	I*4	HABO
3-000000AE	I*4	HABX	3-000000A2	I*4	HARD
2-00000338	I*4	I	AP-00000004@	L*1	LUN
3-00000056	I*4	PARCSR	3-000000AA	I*4	RBLK
3-00000082	I*4	RCNV	3-00000076	I*4	RENQ
3-0000006E	I*4	RNAK	3-0000007E	I*4	RRVI
3-0000009A	I*4	RTIM	3-0000005A	I*4	RXCSR
3-0000005E	I*4	RXDBUF	3-000000A6	I*4	TBLK
3-00000072	I*4	TENQ	3-000000B2	I*4	TEOT
3-0000006A	I*4	TNAK	3-0000007A	I*4	TRVI
3-00000062	I*4	TXCSR	3-00000066	I*4	TXDBUF
3-0000008E	I*4	XACK			

ARRAYS

Address	Type	Name	Bytes	Dimensions
2-000000D1	CHAR	DESCRP	608	(0:18)
3-00000000	L*1	EMB	512	(0:511)
3-00000052	I*4	EMB$L_DV_REGSAV	420	(0:104)
3-00000006	I*4	EMB$Q_HD_TIME	8	(2)
2-00000000	CHAR	OFFSET	209	(0:18)
3-0000006A	I*4	STATS	76	(0:18)

LABELS

Address	Label	Address	Label	Address	Label	Address	Label
1-0000001E	5'	1-00000046	10'	**	20	1-00000069	25'

FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name	Type	Name	Type	Name	Type	Name	Type	Name
I*4	COMPRESS4	I*4	COMPRESSC		DHEAD1		DUP11		DUP3271_QIO		FRCTOF
	IRP$L_PID		IRP$Q_IOSB		IRP$W_BCNT		IRP$W_BOFF	I*4	LIB$EXTZV		LINCHK
	ORB$L_OWNER		UCB$B_ERTCNT		UCB$B_ERTMAX		UCB$L_CHAR		UCB$L_OPCNT		UCB$W_ERRCNT
	UCB$W_STS										

```
0001
0002
0003
0004          subroutine dup3271_qio (lun,emb$w_dv_func)
0005
0006
0007
0008          include 'src$:qiocommon.for /nolist'
0272
0273
0274
0275          byte            lun
0276
0277          integer*2       emb$w_dv_func
0278
0279          integer*4       qiocode(0:1,0:63)
0280
0281
0282
0283
0284          if (qiocode(0,0) .eq. 0) then
0285
0286          Qiocode(1,11) = %loc(IO$_WRITEPBLK)
0287
0288          Qiocode(1,12) = %loc(IO$_READPBLK)
0289
0290          Qiocode(1,13) = %loc(IO$_RDSTATS)
0291
0292          Qiocode(1,26) = %loc(IO$_SETCHAR)
0293
0294          Qiocode(1,29) = %loc(IO$_DIAGNOSE)
0295
0296          Qiocode(1,32) = %loc(IO$_WRITELBLK)
0297
0298          Qiocode(1,33) = %loc(IO$_READLBLK)
0299
0300          Qiocode(1,48) = %loc(IO$_WRITEVBLK)
0301
0302          Qiocode(1,49) = %loc(IO$_READVBLK)
0303
0304          do 10,i = 0,63
0305
0306          qiocode(0,i) = 33
0307
0308          if (qiocode(1,i) .eq. 0) then
0309
0310          qiocode(1,i) = %loc(qio_string)
0311          endif
0312
0313    10    continue
0314          endif
0315
0316          call irp$w_func (lun,emb$w_dv_func,
0317          1 qiocode(0,lib$extzv(0,6,emb$w_dv_func)))
0318
0319          return
0320
```

```
0321            end


PROGRAM SECTIONS

    Name                                Bytes   Attributes

  0 $CODE                                 159   PIC CON REL LCL   SHR   EXE   RD NOWRT LONG
  1 $PDATA                                  8   PIC CON REL LCL   SHR NOEXE   RD NOWRT LONG
  2 $LOCAL                                548   PIC CON REL LCL NOSHR NOEXE   RD   WRT LONG
  3 QIOCOMMON                            1247   PIC OVR REL GBL   SHR NOEXE   RD   WRT LONG

    Total Space Allocated                1962

ENTRY POINTS

    Address  Type  Name

  0-00000000       DUP3271_QIO

VARIABLES

    Address  Type  Name                                Address  Type  Name

 AP-00000008@ I*2  EMB$W_DV_FUNC                     2-00000200  I*4  I
  3-00000442  CHAR IO$_ABORT                         3-0000034D  CHAR IO$_ACCESS
  3-000003C2  CHAR IO$_ACPCONTROL                    3-00000 4B3  CHAR IO$_AVAILABLE
  3-00000297  CHAR IO$_CLEAN                         3-00000369  CHAR IO$_CREATE
  3-00000385  CHAR IO$_DEACCESS                      3-00000393  CHAR IO$_DELETE
  3-0000026D  CHAR IO$_DIAGNOSE                      3-00000065  CHAR IO$_DRVCLR
  3-000004CB  CHAR IO$_DSE                           3-000000A9  CHAR IO$_ERASETAPE
  3-00000276  CHAR IO$_FORMAT                        3-00000071  CHAR IO$_INITIALIZE
  3-00000014  CHAR IO$_LOADMCODE                     3-000003A1  CHAR IO$_MODIFY
  3-000003E2  CHAR IO$_MOUNT                         3-00000000  CHAR IO$_NOP
  3-0000009D  CHAR IO$_OFFSET                        3-000000EB  CHAR IO$_PACKACK
  3-000000E0  CHAR IO$_QSTOP                         3-000003EF  CHAR IO$_RDSTATS
  3-00000421  CHAR IO$_READCSR                       3-00000169  CHAR IO$_READHEAD
  3-000002B6  CHAR IO$_READLBLK                      3-0000013F  CHAR IO$_READPBLK
  3-00000200  CHAR IO$_READPRESET                    3-00000195  CHAR IO$_READTRACKD
  3-0000033A  CHAR IO$_READVBLK                      3-0000045A  CHAR IO$_READWTHBUF
  3-00000484  CHAR IO$_READWTHXBUF                   3-0000004D  CHAR IO$_RECAL
  3-0000007C  CHAR IO$_RELEASE                       3-000001AB  CHAR IO$_REREADN
  3-000001B8  CHAR IO$_REREADP                       3-000000CA  CHAR IO$_RETCENTER
  3-000002E6  CHAR IO$_REWIND                        3-000002C9  CHAR IO$_REWINDOFF
  3-000000FC  CHAR IO$_SEARCH                        3-00000024  CHAR IO$_SEEK
  3-00000231  CHAR IO$_SENSECHAR                     3-00000309  CHAR IO$_SENSEMODE
  3-0000021D  CHAR IO$_SETCHAR                       3-000003B8  CHAR IO$_SETCLOCK
  3-00000088  CHAR IO$_SETCLOCKP                     3-000002DD  CHAR IO$_SETMODE
  3-000002ED  CHAR IO$_SKIPFILE                      3-000002FA  CHAR IO$_SKIPRECORD
  3-00000029  CHAR IO$_SPACEFILE                     3-0000010E  CHAR IO$_SPACERECORD
  3-000003D7  CHAR IO$_STARTDATA                     3-000000B4  CHAR IO$_STARTDATAP
  3-00000037  CHAR IO$_STARTMPROC                    3-0000020F  CHAR IO$_STARTSPNDL
  3-00000059  CHAR IO$_STOP                          3-0000000D  CHAR IO$_UNLOAD
```

```
  3-0000046B  CHAR IO$_WRITEBUFNCRC                3-0000011E  CHAR IO$_WRITECHECK
  3-000001E4  CHAR IO$_WRITECHECKH                 3-000003FF  CHAR IO$_WRITECSR
  3-00000153  CHAR IO$_WRITEHEAD                   3-000002A2  CHAR IO$_WRITELBLK
  3-00000247  CHAR IO$_WRITEMARK                   3-00000314  CHAR IO$_WRITEOF
  3-0000012A  CHAR IO$_WRITEPBLK                   3-000001C9  CHAR IO$_WRITERET
  3-0000017E  CHAR IO$_WRITETRACKD                 3-00000326  CHAR IO$_WRITEVBLK
  3-00000448  CHAR IO$_WRITEWTHBUF                 3-00000257  CHAR IO$_WRTTMKR
 AP-00000004@ L*1  LUN                             3-000004A1  CHAR QIO_STRING
```

ARRAYS

Address	Type	Name	Bytes	Dimensions
2-00000000	I*4	QIOCODE	512	(0:1, 0:63)

LABELS

Address	Label
**	10

FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name
	IRP$W_FUNC	I*4	LIB$EXTZV

COMMAND QUALIFIERS

```
  FORTRAN /LIS=LIS$:DUP3271/OBJ=OBJ$:DUP3271 MSRC$:DUP3271

  /CHECK=(NOBOUNDS,OVERFLOW,NOUNDERFLOW)
  /DEBUG=(NOSYMBOLS,TRACEBACK)
  /STANDARD=(NOSYNTAX,NOSOURCE_FORM)
  /SHOW=(NOPREPROCESSOR,NOINCLUDE,MAP)
  /F77  /NOG_FLOATING  /I4  /OPTIMIZE  /WARNINGS  /NOD_LINES  /NOCROSS_REFERENCE  /NOMACHINE_CODE  /CONTINUATIONS=19
```

COMPILATION STATISTICS

```
  Run Time:            4.83 seconds
  Elapsed Time:        13.61 seconds
  Page Faults:         189
  Dynamic Memory:      190 pages
```

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

CLASSIFY
LIS

DR750
LIS

CSTRING
LIS

DHEADS
LIS

DR780
LIS

DR11W
LIS

COMPRESS
LIS

DTAILS
LIS

DECODECC
LIS

DUMPREG
LIS

DUTLDRIVR
LIS

CALCMAP
LIS

DUP3271
LIS

CRYPTK
LIS

DQDISKS
LIS

DUP11
LIS